



#20

Docket No. 2565-0175P

Application No. 09/337,500

PATENT

IN THE U.S. PATENT AND TRADEMARK OFFICE

In re application of

Before the Board of Appeals

Tomohisa YAMAGUCHI

Appeal No.:

Appl. No.: 09/337,500

Group: 2155

Filed: June 22, 1999

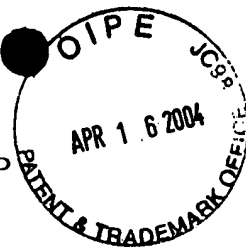
Examiner: T. NGUYEN

For: A SYSTEM OF DYNAMIC MODULE CONFIGURATION
AND A METHOD THEREOF

RECEIVED

APR 21 2004

Technology Center 2100



Docket No. 2565-0175P

Application No. 09/337,500

#20 PATENT

IN THE U.S. PATENT AND TRADEMARK OFFICE

In re application of

Before the Board of Appeals

Tomohisa YAMAGUCHI

Appeal No.:

Appl. No.: 09/337,500

Group: 2155

Filed: June 22, 1999

Examiner: T. NGUYEN

For: A SYSTEM OF DYNAMIC MODULE CONFIGURATION
AND A METHOD THEREOF

RECEIVED

APR 21 2004

Technology Center 2100

TABLE OF CONTENTS

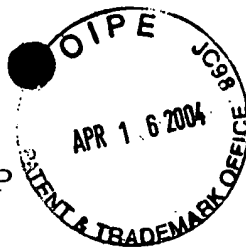
	<u>Page No.</u>
I. Real Party in Interest	2
II. Related Appeals and Interferences.....	2
III. Status of the Claims.....	2
IV. Status of Amendments	2
V. Summary of the Invention.....	3
VI. The Issues.....	6
VII. Grouping of the Claims.....	6
VIII. Argument.....	6
A. Claims 1, 3, 5, 6, 11, 13, 15 and 16-20 are not properly rejected under 35 U.S.C. §103(a) as being unpatentable over Tock in view of Domenikos, et al.	6
1. Brief Description of Applied References (Tock & Domenikos).....	7
2. The References Fail to Teach all the Claimed Limitations	10

Docket No. 2565-0175P

Application No. 09/337,500

B. There is No Motivation to Combine the References.....	15
IX. Conclusion	17
Appendix of Claims	1-9

Docket No. 2565-0175P



Application No. 09/337,500

#20

PATENT

IN THE U.S. PATENT AND TRADEMARK OFFICE

In re application of

Before the Board of Appeals

Tomohisa YAMAGUCHI

Appeal No.:

Appl. No.: 09/337,500

Group: 2155

Filed: June 22, 1999

Examiner: T. NGUYEN

For: A SYSTEM OF DYNAMIC MODULE CONFIGURATION
AND A METHOD THEREOF

RECEIVED

APR 21 2004

Technology Center 2100

PROPOSED BRIEF FOR APPELLANT

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Due: April 16, 2004

Sir:

This appeal is from the decision of the Examiner dated September 17, 2003, finally rejecting claims 1-20, which are reproduced as an Appendix to this Brief. This Brief is being filed in triplicate with the requisite fee.

The commissioner is hereby authorized to charge any appropriate fees under 37 C.F.R. §§1.16, 1.17 and 1.21 that may be required by this paper, and to credit any overpayment, to deposit account 02-2448.

04/19/2004 JBALINAN 00000003 09337500

01 FC:1402

330.00 OP

04/19/2004 JBALINAN 00000004 09337500

01 FC:1251

110.00 OP

I. Real Party in Interest

The named inventors have assigned their rights to the invention that is disclosed in the application and any patent that may issue therefrom to Mitsubishi Denki Kabushiki Kaisha, as recorded in the Patent and Trademark Office at Reel 010062, Frame 0186.

II. Related Appeals and Interferences

To the best of the knowledge of the undersigned, there are no other appeals or interferences known to the Appellants, the Appellant's representatives, or the above noted assignee that will directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

III. Status of the Claims

Claims 1-16, 18 and 20 are currently pending in the application. Claims 1-16, 18 and 20 are rejected and the subject of the appeal. Claims 1, 11, 15 and 18 are independent claims.

IV. Status of Amendments

A response dated November 14, 2003 was filed which included amendments to claims 15 and 18 and the cancellation of claims 17 and 19.

The Examiner has considered these amendments by the advisory action dated December 3, 2003, and has stated that these amendments are entered for the purpose of appeal.

V. Summary of the Invention

In a client device (server) system of the prior art, the client is connected to a device remotely through a network. The device contains execution modules that the client can request to be executed. The device contains memory which is used in the storage and operation of the execution modules. Because the memory being is within the device, the server is limited to the number of requests it can handle and the number of execution modules is offers. The conventional device/server systems also include various hardware components and an operating system in their operations.

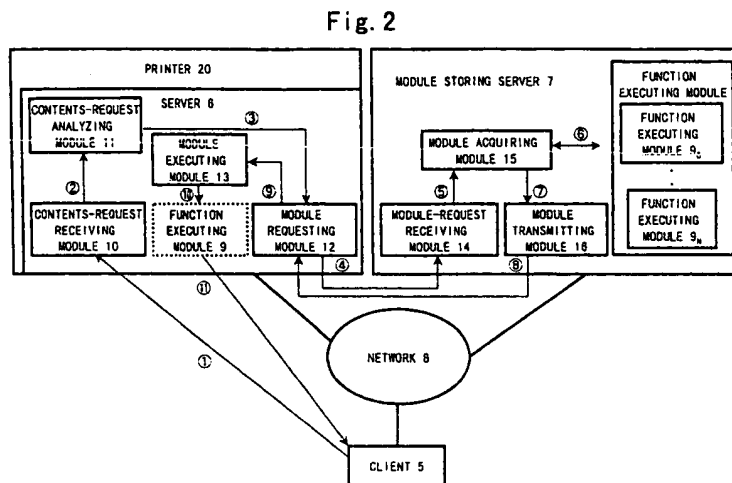
Embodiments of appellant's invention address the above issues resulting from the prior art. The embodiments of appellant's invention provide means which allow devices (e.g., printers, facsimile, television, radio, etc.) connected to a network to limited the memory and other components, and still be able to execute all specific processes requested of the device. See page 4, lines 18-25 to page 5, lines 1-10. This is accomplished by the execution device of appellant's invention located in the device that utilizes the abilities and features, i.e., hardware, operating system, etc. of the device to achieve its

objective. See page 8 of appellant's specification. Further, the devices are connected through the network to a separate memory device. The memory device stores all necessary executable modules which are necessary for executing specific processes. When an execution module is received by a particular device, the requested process is executed and the execution module is then removed from the memory of that device. Thus, this process enables the device to execute a large number of processes without having to store each execution module in the device memory. See page 5, lines 11-25 to page 6, lines 1-20.

Fig. 1 of appellant's invention illustrates the relationship between the three (3) different devices that make up the system defined in appellant's claims. The device 200 includes an execution device 60 that runs the execution modules (9_0 - 9_n) received from memory 70 through the network 8. The request device 50 allows a user to remotely connect to the execution device and request execution of a function execution module stored on the remote memory. See pages 4, lines 18-25 to page 6, lines 1-20.

Fig. 2 of appellant's invention illustrates appellants system as it operates with a printer 20 acting as the device. The server 6 acts as the execution device 60 described with reference to Fig. 1. The module storing server 7 functions as the memory 70, and stores execution modules necessary for server 6 to operate in conjunction with the client's 5 requests. Each of the

printer 20, storing server 7 and client 5 are remotely located from each other and in communication through the network 8. See pages 6, lines 21-25 to page 7, lines 1-20. Fig. 2 has been reproduced below to aide in understanding the present invention.



The unique configuration of appellant's system allows the device 200 to include the bare necessity of circuitry to operate. This allows the device the flexibility of being smaller, easier to manufacture, more versatile. More importantly, appellant's invention creates a device that is not limited to the number of client requests nor is it limited to the number of execution modules stored within the device itself.

VI. The Issues

The final Office Action presents one issue for review on Appeal.

1. Whether claims 1, 3, 5, 6, 11, 13, 15 and 16-20 are properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Tock (U.S. Patent No. 5,815,718) in view of Domenikos et al. (U.S. Patent No. 5,838,810).

VII. Grouping of the Claims

For purpose of this appeal, Appellants group the claims as follows:

- A) Independent claims 1 and 11 and their dependent claims to stand or fall together
- B) Independent claims 15 and 18 and their dependent claims stand or fall together

VIII. Argument

- A. Claims 1, 3, 5, 6, 11, 13, 15 and 16-20 are not properly rejected under 35 U.S.C. §103(a) as being unpatentable over Tock in view of Domenikos, et al.

To establish *prima facie* obviousness, all claim limitations must be taught or suggested by the prior art and the asserted modification or combination of prior art must be supported by some teaching, suggestion, or motivation in the applied reference or in knowledge generally available to one skilled in the art.

In re Fine, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988); *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992). Thus, “[a]ll words in a claim must be considered in judging the patentability of that claim against the prior art.” *In re Wilson*, 424 F.2d 1382, 1385, 165 USPQ 494, 496 (CCPA 1970). The prior art must suggest the desirability of the modification in order to establish a *prima facie* case of obviousness. *In re Brouwer*, 77 F.3d 422, 425, 37 USPQ2d 1663, 1666 (Fed. Cir. 1995). It can also be said that the prior art must collectively suggest or point to the claimed invention to support a finding of obviousness. *In re Hedges*, 783 F.2d 1038, 1041, 228 USPQ 685, 687 (Fed. Cir. 1986); *In re Ehrreich*, 590 F.2d 902, 908-09, 200 USPQ 504, 510 (CCPA 1979).

1. Brief Description of Applied References (Tock & Domenikos)

Tock

Tock discloses a method and system that when applying object oriented programming, limits the amount of random access memory (RAM) that is used. This is accomplished by partitioning an application into two (2) separate address spaces using an offline class loader 132. The first address space resides in a read only memory device (ROM) 208 and contains methods that do not require dynamic loading and data that remains constant. The second address space resides in a read/forward write memory device (RAM) 210 and

contains the methods that require dynamic loading and data that is changed during execution. See column 3, lines 1-25 to column 5, lines 19-37.

During dynamic loading data that resides on one or more servers 108 within a computer network 106 can be referenced, acquired and executed by a computer 114 connected to the servers 108. This data is then stored in one of the two partition addresses as described above. The computer 114, upon retrieving the data, uses this data in execution of a program. The system of Tock locates and retrieves data, acquires the data, separates the data into one of two memory locations based on the type of data and executes a program using the data.

Further, in the system of Tock, a client computer 102 is connected through a network to one or more servers 108. Within each of these servers 108 are processors 112, communication interfaces 116, operating systems 118 and memory 110 for storing various class files, etc. as illustrated in Fig. 1. The client computer 102 when executing a browser, provides the user of the client computer 102 with access to files located on the server 108. See col. 3, lines 53-67 through column 4. In essence, the network 106 acts as a bridge between the client's computer 102 and client server 108. Everything necessary for the client computer is contained within the server 108. Thus, each server 108 maintains its own extensive memory for storing files.

Domenikos

Domenikos is similar in operation to that of Tock in which two separate devices are located through a network to communicate with each other. This is contrary to appellant's invention which provides three separate devices, for example, a request device and an execution device and a memory. Each of these three devices are connected to a network. The three devices operate together through the network to request an execution module, retrieve the execution module and execute the module.

Domenikos provides a system in which two devices operate together through a network to find, retrieve and execute applications. In Domenikos, a client (12) accesses through a network an Internet server (14). See column 8, lines 5-35. The Internet server contains various programs for use on different platforms such as a PC (50), UNIX (52) or MAC (54). See column 10, lines 36-41. The application program may then be executed at the remote server location during a browser operation. See column 12, lines 15-67 through column 13, lines 1-3 and column 14 lines 39-56. The system of Domenikos therefore operates by the client sending a request to the server and the server retrieving and executing the requested application.

2. The References Fail to Teach all the Claimed Limitations

Tock teaches exactly what the present invention has been designed to overcome. Appellant's invention alleviates the necessity to have a large memory and store large amounts of information within a device in connection with a request from a client computer. In contrast to Tock, appellant's invention utilizes a device, i.e., a printer, scanner, etc. which has embedded therein an execution device, which performs server functions for obtaining and executing function modules. This execution device does not function like conventional servers. Conventional servers, as discussed in the summary above, include various hardware and an operating system in their operations, however, the execution device of appellant's invention is installed in a device and utilizes the abilities and features, i.e., hardware, operating system, etc. of the device to achieve its objective. See page 8 of appellant's specification. When a request from the request device is sent to the execution device, the execution device sends a request over the network to a memory remotely located on the network, which stores function modules. The correct function module is selected and provided back to the execution device through the network. The function module is then executed based on the request and the result is transferred to the request device through the network. The separate memory stores all the necessary information to perform numerous tasks. This reduces the necessity for memory within each of the devices themselves, which

provides a greater advantage over other systems that operate using only function modules stored within the device, by, the reduced memory, allowing the devices to be manufactured smaller and cheaper, and providing a greater number of execution modules which creates greater flexibility and abilities for the devices.

The features the Examiner has relied upon do not teach or anticipate the claimed features as recited in claims 1, 11, 15 and 18. The Examiner alleges that item 110 of Fig. 1 corresponds to the claimed memory. Memory 110 is located within the server, thus Tock provides a system similar to the conventional systems discussed in the summary above on which the present invention improves upon. Having the memory located within the server itself limits Tock's system to providing only those function modules that can be stored in the internal memory and therefore, Tock's system lacks the advantages associated with appellant's invention as discussed above.

The Examiner alleges that the client computer 102 of Tock corresponds to the claimed request device. In the language of the independent claims, claim 1 recites "a request device, located on said network remotely from said memory and request device, which outputs an execution request for executing one of the specific processes; and an execution device for receiving through the network the execution request output".

Claim 11 recites “outputting, by a request device through a network, an execution request for executing one of the specific processes; and receiving, by an execution device, located on said network remotely from said memory and said request device, the execution request through the network”.

Claim 15 recites “receiving an execution request, through a network from a request device, which requests a performance of a function of the device ... wherein the external resource is located remotely on said network from said execution device.”

Claim 18 recites “requesting an action by a request device to be performed by a selected operational device which is achieved through a set of instructions contained in a requested execution module ... wherein the request device is located remotely on the network from said plural operational devices and said memory”.

As recited in appellant’s independent claims, the execution request is sent through the network to the execution device. The Examiner alleges that the abstract in column 3, lines 46-49 discloses such features. However, the abstract only discloses the storing and identifying of executable module data. Column 3, lines 46-49 disclose the execution of an executable module using a java interpreter but fails to disclose the unique combination of obtaining a request from a request device through a network by the execution device and

further obtaining by the execution device the execution module through the network from a remotely located memory.

Further, the Examiner alleges that column 5, lines 20-25 disclose the acquiring and execution of the execution module through the network. However, this section of text merely describes the relationship of various features within memory 110, of which the Examiner refers to as corresponding to appellant's claimed memory. First, this relationship, as disclosed in Tock, is not performed through the network, but is performed within the various elements in memory 110 of Tock.

Second, in appellant's claimed invention, the function executing module which requests the execution modules from a memory is not located within the memory from which the modules are obtained. The function executing module and memory of appellant's invention are remotely located on the network from each other. The Examiner, however, is arguing that the same memory device (110) within Tock, which he alleges corresponds to appellant's memory, also corresponds to the execution device claimed by appellant. As stated above, however, appellant's execution device and memory are remotely located on the network from each other.

The Examiner asserts that Domenikos "teaches an execution device located on said network remotely from said memory and said request device". Appellants disagree. First, Domenikos does not teach the use of an execution

device, memory and request device, as claimed. To the contrary, Domenikos teaches the utilization of two (2) separate devices an agent/client (12) and a server (14). Appellant concedes that these two devices communicate with each other remotely through a network. However, appellant's claimed invention combines three (3) separate devices, namely the execution device, memory and request device, each of which are remote from each other and connected through a network.

Furthermore, in regard to claims 15 and 18, each of these claims further define by reciting, beyond the three (3) separated devices noted above not taught by Tock and Domenikos, the integration of the request device in a device itself. Claim 15 recites, "an internal resource of a device for performing an original function of the device, and ... receiving an execution request, through the network from a request device, which requests a performance of a function of the device ... wherein the receiving, acquiring and executing are performed by using a part of the internal resource".

Claim 18 recites, "storing plural diverse execution modules in a memory located remotely on said network from said operational devices, each of said execution modules containing a set of instructions usable by an operational device; requesting an action by a request device to be performed by a selected operational device which is achieved through a set of instructions contained in a requested execution module; acquiring said requested execution module by

said selected operational device from said remote memory, said operational device executing said set of instructions contained in said requested execution module to perform the requested action”.

Both Tock and Domenikos, only teach the implementation of their system within a server and not a separate device, such as a printer, personal digital assistant (PDA) etc. Thus, not only does Tock and Domenikos fail to teach or suggest the use of the three (3) separate devices of appellant's claimed invention as discussed above, but they also fail to teach or suggest implementing such a system to be used in connection with various devices as claimed by appellant.

It is apparent that neither Tock or Domenikos teach each and every feature of the claimed invention, alone or in combination. The distinct features of appellants claimed invention which includes three (3) separate devices is not taught by the references. Thus, the Examiner has failed to establish a combination of references that teaches all the claimed features as required under a 35 U.S.C. 103 rejection.

B. There is no Motivation to Combine the References

The Examiner's rejection further fails to provide motivation for the Examiner's alleged combination. In order for a *prima facie* case to exist, the prior art must suggest the desirability of the claimed invention, providing

motivation to make the combination proposed by the Examiner. *In re Rouffet*, 149 F.3d 1350, 1357, 47 USPQ 2d 1453, 1457-58 (Fed.Cir. 1998). The level of skill in the art cannot be relied upon to provide this suggestion to combine the references. *Al-Site Corp. v. VSI Int'l Inc.*, 174 F.3d 1308, 50 USPQ 2d 1161 (Fed.Cir. 1999).

The Examiner has failed to provide proper motivation to combine the teachings of Tock with those of Domenikos. The Examiner suggests that one of ordinary skill in the art would have been motivated to “combine the teachings of Tock and Domenikos to have an execution device located on said network remotely from said memory and said request device because it would increase the speed of execution of application remotely and reduce the storing load on server.” The Examiner’s reasoning does not make much sense, since both Tock and Domenikos teach the inclusion of memory within the server and thus it would be necessary to extensively modify both systems to achieve appellant’s claimed invention.

As stated in MPEP § 2143.01, if the proposed modification or combination of the prior art would change the principle of operation of the prior art invention being modified, then the teachings of the references are not sufficient to render the claims prima facie obvious. *In re Ratti*, 270 F.2d 810, 123 USPQ 349 (CCPA 1959). Thus, because extensive modification of Tock’s and Domenikos systems would be required to accomplish the claimed

invention, one of ordinary skill would not look to these two references to provide teachings that would suggest or teach appellant's invention. Therefore, any teachings of Tock and Domeniko's used in a rejection of appellant's claims would not be an obvious inclusion.

Further, neither Tock nor Domenikos suggest modifying their systems such that they utilize a request device, execution device and memory remotely in the manner claimed by appellant. In fact, as stated above, the combination of Tock and Domenikos fails to teach all the claim limitations and thus one of ordinary skill would not be motivated to combine these two references to achieve appellant's claimed invention. More simply, motivation to combine the references to achieve appellant's claimed invention can't exist because the combination lacks the necessary features to achieve appellant's invention as claimed.

IX. Conclusion

Based on the reasons set forth above, the rejections of claims 1-20 under 35 U.S.C. §103 should be REVERSED. As shown in the foregoing arguments, the claimed features of the present invention are not disclosed or suggested in the cited documents. Further, one of ordinary skill in the art would not look to combine the teachings of the references. Accordingly, reversal of the rejection is respectfully requested.

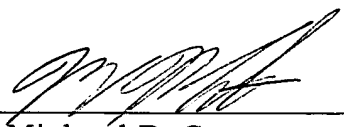
Docket No. 2565-0175P

Application No. 09/337,500

If necessary, the Commissioner is hereby authorized in this, concurrent, and future replies, to charge payment or credit any overpayment to Deposit Account No. 02-2448 for any additional fees required under 37 C.F.R. 1.16 or under 37 C.F.R. 1.17; particularly, extension of time fees.

Respectfully submitted,

BIRCH STEWART KOLASCH & BIRCH, LLP

By: 
Michael R. Cammarata
Registration No. 39,491

MRC:CJB/cb
2565-0175P

P.O. Box 747
Falls Church, VA 22040-0747
(703) 205-8000

Appendix of Claims

Claim 1.

A system of dynamic module configuration which is linked through a network comprising:

a memory, linked to the network, for storing a plurality of function executing modules which execute specific processes;

a request device, located on said network remotely from said memory, which outputs an execution request for executing one of the specific processes; and

an execution device, located on said network remotely from said memory and said request device, for receiving, through the network, the execution request output from the request device, acquiring, through the network, one of the plurality of function executing modules which has a function of realizing the execution request from the memory, executing the acquired function execution module and providing a result of the execution of the function execution module to the request device.

Claim 2.

The system of dynamic module configuration of claim 1, wherein the execution device deletes the acquired function execution module after the acquired function execution module has been executed.

Claim 3.

The system of dynamic module configuration of claim 1, wherein the execution device stores the acquired function execution module after the acquired function execution module has been executed, and re-executes the acquired function execution module stored in the execution device when it is requested to execute a module having a function corresponding to the acquired function execution module.

Claim 4.

The system of dynamic module configuration of claim 1, wherein the memory caches the function execution module acquired by the execution device and provides the function execution module cached in the memory when it is requested to acquire a module, which has a function corresponding to the function execution module cached in the memory, by the execution module.

Claim 5.

The system of dynamic module configuration of claim 1, wherein the request device and the memory are installed in a device.

Claim 6.

The system of dynamic module configuration of claim 1, wherein the request device is a client which outputs a contents request corresponding to the execution request, the execution device is a server which receives the contents request and responds to the contents request, and the memory is a module storing server which stores the plurality of function executing modules for responding to the contents request.

Claim 7.

The system of dynamic module configuration of claim 6, wherein the server includes

a contents-request receiving module for receiving the contents request from the client,

a contents-request analyzing module for analyzing the contents request received by the contents-request receiving module in order to select one of the plurality of function executing modules which has a function needed in responding to the contents request,

a module requesting module for requesting a selected function executing module from the module storing server based on an analyzing result by the contents-request analyzing module, and for receiving the selected function executing module from the module storing server, and

a module executing module for executing the selected function executing module received by the module requesting module.

Claim 8.

The system of dynamic module configuration of claim 7, wherein the module storing server includes

a module-request receiving module for receiving a module request from the module requesting module,

a module acquiring module for acquiring a function executing module out of the plurality of function executing modules based on the module request received by the module-request receiving module, and

a module transmitting module for transmitting the function executing module acquired by the module acquiring module to the server.

Claim 9.

The system of dynamic module configuration of claim 7, wherein the server further includes a module storing module for storing the selected

function executing module acquired from the module storing server as many as possible in a resource of the server.

Claim 10.

The system of dynamic module configuration of claim 8, wherein the module storing server further includes a module caching module for caching the selected function executing module after the selected function executing module has been sent to the server.

Claim 11.

A dynamic module configuration method using a network comprising the steps of:

storing in a memory a plurality of function executing modules for executing specific processes;

outputting, by a request device through the network, an execution request for executing one of the specific processes; and

receiving, by an execution device, located on said network remotely from said memory and said request device, the execution request through the network, acquiring, through the network, one of the plurality of function executing modules from the memory which has a function of realizing the execution request, executing the acquired function execution module and

providing a result of the execution of the function execution module to the request device.

Claim 12.

The dynamic module configuration method of claim 11, wherein the step of executing the acquired function execution module includes the step of deleting the acquired function execution module after the acquired function execution module has been executed.

Claim 13.

The dynamic module configuration method of claim 11, wherein the step of executing the acquired function executing module includes the step of storing the acquired function execution module after the acquired function execution module has been executed, and re-executing the acquired function execution module when it is requested to execute a module having a function corresponding to the acquired function execution module.

Claim 14.

The dynamic module configuration method of claim 11, wherein the step of memorizing the plurality of function executing modules includes the step of caching the acquired function execution module, and providing the acquired function execution module cached at the caching step when it is requested to

acquire a module having a function corresponding to the acquired function execution module.

Claim 15.

A system of dynamic module configuration comprising:

an internal resource of a device for performing an original function of the device; and

an execution device for

receiving an execution request, through a network from a request device, which requests a performance of a function of the device,

acquiring, from an external resource, one of a plurality of function execution modules which has a function of realizing the execution request, and executing the acquired function execution module,

wherein the receiving, acquiring and executing are performed by using a part of the internal resource and wherein an executed result is obtained from executing the function execution module and the result is provided to the device; and

wherein the external resource is located remotely on said network from said execution device.

Claim 16.

The system of dynamic module configuration of claim 15, wherein the internal resource includes a central processing unit and a memory, the execution device includes a program stored in the memory and executed by the central processing unit, and the external resource includes a memory, being independent of the device, for memorizing the plurality of function execution modules.

Claim 18.

A method of providing execution module instructions to plural operational devices on a network, comprising the steps of:

storing plural diverse execution modules in a memory located remotely on said network from said operational devices, each of said execution modules containing a set of instructions usable by an operational device;

requesting an action by a request device to be performed by a selected operational device which is achieved through a set of instructions contained in a requested execution module;

acquiring said requested execution module by said selected operational device from said remote memory, said operational device executing said set of instructions contained in said requested execution module to perform the requested action; and

Docket No.: 2565-0175P

Appln. No.: 09/337,500

wherein the request device is remotely located on the network from said plural operational devices and said memory.

Claim 20.

The method of claim 18, wherein the operational device includes an execution device for executing the requested execution module acquired from said memory.